

REMARKS

Claims 1-57 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Objection to the Title:

The Examiner objects to the title as not being descriptive and indicative of the invention to which the claims are directed. The Examiner suggests adding "CORBA" to the title. Applicants submit that the present title, "Secure Access to Managed Network Objects using a Configurable Platform-Independent Gateway" is descriptive and indicative of the invention to which the claims are directed. The invention is not limited to only CORBA embodiments. Therefore, adding the word "CORBA" to the title would in fact misrepresent the present invention.

Information Disclosure Statement:

Applicants note that an information disclosure statement was submitted electronically and via mail with an accompanying Form PTO-1449 on November 12, 2003. Applicants respectfully request the Examiner to carefully consider the listed references and return copies of the signed and initialed electronic submission and Form PTO-1449.

Section 102(e) Rejection:

The Office Action rejected claims 1-57 under 35 U.S.C. § 102(e) as being anticipated by Barker et al. (U.S. Patent 6,363,421) (hereinafter "Barker"). Applicants respectfully traverse this rejection in light of the following remarks.

Regarding claim 1, the Examiner states, "Barker teaches ... wherein the gateway is configured to provide object-level access control between the managers and the

managed objects to send the requests to the managed objects....” Applicants respectfully disagree with the Examiner’s interpretation of Barker.

Applicants submit that Barker does not anticipate a gateway that is configurable to provide object-level access control between the managers and the managed objects. Barker discloses a system for “access control based on client name and password” (Barker, column 8, lines 45-46). Barker describes this as “a method of *client based* access control of network elements...” (Emphasis added) (Barker, column 30, lines 45-46). Further, Barker summarizes his access control features with “the client based access control described above provides a means to restrict access on a command/client basis” (Barker, column 31, lines 10-12).

The Examiner has cited a passage from Barker (column 23, line 55 – column 26, line 10) describing a set of procedures whereby client may register to receive notification when managed object attribute values change. The Examiner specifically quotes one line stating, “[n]ote that if more than one attribute has changed for a managed object instance, the changes will be grouped and delivered to each registered client on a managed object instance basis” (Barker, column 26, lines 6-10). Although this portion of Barker teaches clients receiving attribute updates from individual managed objects, it does not teach object-level *access control*. Barker explicitly teaches providing client based access control at the start of a session (see Barker, column 30, lines 47-52), while not requiring further authentication or access control based upon which managed objects the client wishes to access.

As the Examiner has stated, “[e]ach managed object class requires the session identifier as a parameter to each public method” (Barker, column 30, lines 56-58). Applicants assert that including a session identifier as a parameter in each public method only allows a managed object class to validate the current session – i.e. to ensure that the client has registered with the server and that the session is currently valid. Barker does not teach a client presenting a user name, password or other authentication credentials when registering for object attribute update notification. Instead, Barker teaches that a

client must only provide the session ID, object instance identifier, a set of desired attribute codes, and a callback function when registering for attribute update notifications (see Barker, column 25, lines 23-30).

Additionally, Barker teaches that a client can specify a range of managed object instance identifiers, or even request all instances in a managed object call through the managed object instance identifier parameter (Barker, column 25, lines 27-28). Hence, Barker teaches that once a client has been properly authenticated at the start of a session, that client may then register for attribute update notification for a number of managed objects through a single call. Such functionality is clearly not compatible with object-level access control and thus, Barker clearly teaches away from object-level access control.

The Examiner also cites Barker teaching, “access permissions associated with the session are examined before authorizing client execution (e.g. remove operation)” (parenthesis in original) (Barker, column 30, lines 58-60). However, this portion of Barker is clearly referring to ensuring that the client has started a valid session with the server. In fact, Barker, referring to the same remove operation, clearly states, “[a]s with any other client requests, the *client must have created a session prior to performing this operation.*” (Emphasis added) (Barker, column 22, lines 51-53).

Thus, Applicants assert that Barker does not teach object-level access control between the managers and the managed objects. For at least the reasons given above, the rejection of claim 1 is not supported by the prior art and its removal is respectfully requested.

Regarding claim 20, the Examiner contends that “Barker teaches... wherein the gateway is configured to ... determine on a managed object level whether or not the manager application is allowed to send a request to the managed object as a function of the user of the manager application.” Applicants disagree with the Examiner’s interpretation of Barker.

Barker fails to anticipate determining on a managed object level whether or not the manager application is allowed to send a request to the managed object. In contrast, as shown in the arguments regarding claim 1 above, Barker discloses a method of client based access control of network elements as a means to restrict access on a command/client basis.

Further, Barker teaches the use of a single service object “to provide services for a class of managed objects” (Barker, column 14, lines 42-43) and that the EM server “will implement one application-specific service object for each type of physical or logical resource to be managed” (underlining added) (Barker, column 39, lines 60-62). Applicants assert that access control on a command/client basis while using a single service object for each class of managed object actually teaches away from determining on a managed object level whether or not the manager application is allowed to send a request to the managed object.

For at least the reasons given above, the rejection of claim 20 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 20 apply to claim 39.

Regarding claim 2, the Examiner states, “Barker teaches ... the gateway is configurable to determine whether each of the managers is authorized to communicate with each of the managed objects....” Applicants respectfully disagree with the Examiner’s interpretation of Barker.

Barker teaches the use of a single service object “to provide services for a class of managed objects” (Barker, column 14, lines 42-43) and that the EM server “will implement one application-specific service object for each type of physical or logical resource to be managed” (underlining added) (Barker, column 39, lines 60-62). Applicants assert that access control on a command/client basis while using a single service object for each class of managed object actually teaches away from determining

on a managed object level whether or not the manager application is allowed to send a request to the managed object.

Further, Barker discloses client based access control that provides a means to restrict access on a command/client basis (Barker, column 31, lines 10-12). Hence, Barker teaches access control based on a command/client basis, not a managed object basis and thus fails to disclose a gateway that is configurable to determine whether each of the managers is authorized to communicate with each of the managed objects.

For at least the reasons given above, the rejection of claim 2 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 2 apply to claims 21, and 40.

Regarding claim 3, Barker fails to teach a gateway that is configurable to authenticate the managers to receive the events from or to send the request to the managed objects as a function of the identity of the managed object as the Examiner asserts. As the Examiner states, Barker teaches the use of basic server authentication, SSL, and web server administration including client name and password for access control (Barker, column 8, lines 31-54). Further, Barker discloses client based access control that provides a means to restrict access on a command/client basis (Barker, column 31, lines 10-12). However, Applicants can find no reference in Barker regarding a gateway that is configurable to authenticate the managers to receive the events from or to send the request to the managed objects as a function of the identity of the managed object.

For at least the reasons given above, the rejection of claim 3 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 3 apply to claims 22, and 41.

Regarding claim 8, the Examiner states, “Barker teaches ... the managed objects comprise one or more objects corresponding to a telephone network....” Applicants respectfully disagree with the Examiner’s interpretation of Barker.

Barker discloses that the system client is connected to a network element and element management system client through a public switched telephone network (Barker, column 3, lines 48-53). Additionally, Barker teaches the use of a telephone system network through the computer internet and a telephonic link for a system client to connect to the system server (Barker, column 3, lines 54-62). Hence, Barker discloses using a telephone connection between clients and servers. However, Applicants can find no reference in Barker regarding managed objects comprising one or more objects corresponding to a telephone network. None of the managed objects in Barker correspond to a telephone network.

For at least the reasons given above, the rejection of claim 8 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 8 apply to claims 27, and 46.

Regarding claim 10, the Examiner contends that Barker teaches a gateway that is configurable to provide security audit trails. Applicants disagree with the Examiner.

Applicants submit that at the Examiner’s cited passage (Barker, column 30, lines 44-63), regarding the server retrieving the client record from local data services and returning a session object to the client, Barker is referring to loading a client record (e.g. a record in a database) containing authentication information from a datastore, and does not relate to security audit trails.

The Examiner’s other cited pages regarding the event distributor refer to a component that “provides event routing and distribution” (underlining added) (Barker, column 11, lines 21-22). Applicants submit that this event routing and distribution is

performed to communicate events from sender to recipient and has nothing to do with security audit trails as asserted by the Examiner.

The Examiner further contends that the Client Session Manager auditing for sessions that have that have terminated without notifying the manager implies that Barker teaches the gateway is configurable to provide security audit trails. Applicants disagree with the Examiner's interpretation of Barker's Client Session Manager.

Applicants submit that Barker defines auditing as the periodic polling of configuration data and persistent attributes (Barker, column 19, lines 42-47). Thus, under Barker, the Client Session Manager periodically polls "active sessions/applications to see if any session/application has failed to check in recently" (Barker, column 16, lines 62-63). Thus, the Client Session Manager does not provide security audit trails as the Examiner contends. Therefore, Applicants assert that Barker fails to teach that a gateway is configurable to provide security audit trails.

For at least the reasons given above, the rejection of claim 10 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 10 apply to claims 29, and 48.

Regarding claim 11, the Examiner states that Barker teaches the gateway providing security audit trails comprises the gateway providing access to a logging service. Applicants respectfully disagree with the Examiner.

As shown in the arguments above regarding claim 10, Barker fails to teach a gateway providing security audit trails. Barker also fails to teach the gateway providing security audit trails comprises the gateway providing access to a logging service.

For at least the reasons given above, the rejection of claim 11 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 11 apply to claims 30, and 49.

Regarding claim 12, the Examiner contends that Barker teaches, “the logging service (local data services at the server) is operable to log an ID of a user that sends each request” (parenthesis and underlining in original). Applicants respectfully disagree with the Examiner’s interpretation of Barker.

As shown above in the arguments regarding claim 10, the local data services described by Barker do not provide security audits comprising providing access to a logging service.

Further, the Examiner cites Barker stating, “a client application must register with the server by providing identification of the client host, port, client, and a password” (Barker, column 30, lines 48 – 50). Applicants can find no mention in the cited passage, nor in the entirety of Barker, regarding the ID of a user that sends each request. Applicants maintain that not only does such registration fail to provide a logging service that is operable to log an ID of a user, Barker does not include the ID of a user that sends each request in the registration process.

For at least the reasons given above, the rejection of claim 12 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 12 apply to claims 31, and 50.

Regarding claim 18, the Examiner states, “Barker teaches ... the requests are converted from the interface definition language to a Portable Management Interface (PMI) format prior to delivery to the managed objects”. Applicants respectfully disagree with the Examiner’s interpretation of Barker.

Barker teaches, “SNMP Mediator 160 provides translation between the MIB ASN.1 format and the managed object notation used in this architecture” (Barker, column 11, lines 39-42). The examiner contends that this translation constitutes converting from the interface definition language to a Portable Management Interface (PMI) format.

Applicants assert the Portable Management Interface is a specific interface that is not the managed object notation used by Barker. Further, Applicants can find no reference in Barker regarding a Portable Management Interface (PMI) format. Therefore Applicants submit that Barker fails to teach that the requests are converted from the interface definition language to a Portable Management Interface (PMI) format prior to delivery to the managed objects as contended by the Examiner.

For at least the reasons given above, the rejection of claim 18 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 18 apply to claims 37, and 56.

Regarding claim 19, the Examiner states, “Barker teaches ... the requests are converted from the interface definition language to a platform-specific format prior to delivery to the managed objects,” citing Barker’s SNMP mediator providing translation between the MIB ASN.1 format and the managed object notation used in this architecture. Applicants respectfully disagree with the Examiner’s interpretation of Barker.

Barker teaches the use of SNMP as the communication protocol between element management system and the managed elements (Barker, column 4, lines 43-45). Applicants assert the SNMP is not a platform-specific format, but rather is a network protocol that contains no platform specific features. Thus, Barker fails to teach the requests are converted from the interface definition language to a platform-specific format prior to delivery to the managed objects as asserted by the Examiner.

For at least the reasons given above, the rejection of claim 19 is not supported by the prior art and its removal is respectfully requested. Similar remarks as discussed above in regard to claim 19 apply to claims 38, and 57.

Applicants also assert that numerous other ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been

shown to be patentably distinct, a further discussion of the dependent claims is not required at this time.

CONCLUSION

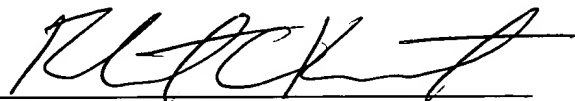
Applicants submit the application is in condition for allowance, and an early notice to that effect is requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-48400/RCK.

Also enclosed herewith are the following items:

- ☒ Return Receipt Postcard
- ☐ Petition for Extension of Time
- ☐ Request for Approval of Drawing Changes
- ☐ Notice of Change of Address
- ☐ Fee Authorization Form authorizing a deposit account debit in the amount of \$
for fees ().
- ☐ Other:

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: April 8, 2004